

# Sesión 1: Procesos

## Hoja de problemas


### Concurrencia

Ángel Herranz

aherranz@fi.upm.es

Universidad Politécnica de Madrid

2019-2020

 **Ejercicio 1.** Lee la guía docente de la asignatura disponible en Moodle:


[https://www.upm.es/comun\\_gauss/publico/guias/2019-20/2S/GA\\_10II\\_105000018\\_2S\\_2019-20.pdf](https://www.upm.es/comun_gauss/publico/guias/2019-20/2S/GA_10II_105000018_2S_2019-20.pdf)

 **Ejercicio 2.** Antes de la primera entrega de los ejercicios cortos, es importante leer


<https://docs.oracle.com/javase/tutorial/essential/concurrency/index.html>

y en concreto la parte relacionada con las instancias de *Thread*:

<https://docs.oracle.com/javase/tutorial/essential/concurrency/threads.html>

 **Ejercicio 3.** Andrews y Schneider, en 1983 escribieron un artículo sobre los conceptos fundamentales de la concurrencia: *Concepts and Notations for Concurrent Programming*. Decadas después, ese artículo sigue siendo vigente. Emplea las dos primeras semanas del cuatrimestre a leer las secciones 1, 2, 3.1 y 3.2 del artículo:

[http://babel.ls.fi.upm.es/teaching/concurrencia/material/concepts\\_and\\_notations.pdf](http://babel.ls.fi.upm.es/teaching/concurrencia/material/concepts_and_notations.pdf)

 **Ejercicio 4.** Experimenta con el programa HolaMundos de las transparencias. Introduce bucles, haz que los procesos *digan más cosas* por la salida estándar (identificadores de objeto, identificadores de proceso, pasos por sus bucles, etc.)<sup>1</sup>, juega con `start` y `join`.

---

<sup>1</sup>Explora los métodos de la clase `Thread`, por ejemplo los métodos de clase `currentThread` y `sleep`, o el método de instancia `getId`.